[**MYSQL查询语句优化**](http://www.cnblogs.com/mliang/p/3637937.html)

mysql的性能优化包罗甚广： 索引优化，查询优化，查询缓存，服务器设置优化，操作系统和硬件优化，应用层面优化（web服务器，缓存）等等

**查询开销指标：**

执行时间，检查行数，返回的行数

**建立索引的几个准则：**

1：合理的索引，2索引越多更新数据的速度越慢，3尽量采用MyIsam 作为引擎的时候使用索引（MyIsam不支持Transcation）4在达到性能极限时，使用memcached，4EXPLAIN分析SQL语句

**1. count的优化**

a. select count(\*) from world.city where id > 5;

b. select (select count(\*) from world.city) – count(\*) from world.city where id <= 5;

**2. 避免使用不兼容的数据类型。**

Float 与int不兼容 varchar、binary和varbinary是不兼容的。数据类型的不兼容可能是优化器无法执行一些本来可以优化的操作。

保证在功能的实现的基础上，尽量减少对数据库的访问次数；通过搜索参数，尽量减少对表的访问行数，最小化结果集，从而剑气网络负担，读写分离，提高访问访问速度，尽量将使用的索引放在首列，尽可能减少结果集，按照特定顺序提取数据

**3. 索引字段上进行运算会使索引失效。**

尽量避免在WHERE 自居中对字段进行函数式或表达式操作，如

Select \* from t1 where F1/2=100 改成 select \* from t1 where F1=100\*2

**4. 避免使用!=或＜＞、IS NULL或IS NOT NULL、IN ，NOT IN等这样的操作符.**

这会使索引失效

如：select \* from employee where id ！=“B%”，无法通过索引来确定将要命中的行数，因此需要搜索该表的所有行 尽量使用exists代替in

**5. 尽量使用数字型字段.**

这会减少存储开销

**6. 合理使用EXISTS,NOT EXISTS子句。如下所示：**

1：SELECT SUM(T1.C1) FROM T1 WHERE (SELECT COUNT(\*) FROM T2 WHERE T2.C2=T1.C2>0 )

2：SELECT SUM(T1.C1) FROM T1 WHERE EXISTS(SELECT \* FROM T2 WHERE T2.C2=T1.C2)

效果相同，后者的效率高于前者，

**7. 能够用BETWEEN的就不要用IN**

**8. 能够用DISTINCT的就不用GROUP BY**

**9. 尽量不要用SELECT INTO语句。SELECT INTO 语句会导致表锁定，阻止其他用户访问该表。**

**10. 必要时强制查询优化器使用某个索引**

**索引分四类**

**Index-----普通索引，数据可以重复**

**Fulltext------全文索引，用来对大标的文本域进行索引**

**Unique----唯一索引，要求记录唯一**

**Primary key ---主键索引，唯一主键的列上必须是主键**

**like索引的使用**

**“XXXX%”**

**原则 ：**单表数据太少，可以不加索引

Where后的条件，order by group by 后的列最好适当加上索引

联合查询 子查询等多表操作

ps：数据量特别大的时候，最好不要用联合查询，即使你做了索引